IN THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method of sharing secure cryptographic connections between trusted computing entities which share a secret value, the computer-implemented method comprising the steps of:

connecting an originally-connected entity to an original endpoint, the originally-connected entity having an entity name and cryptographic context information; and

creating an entity identifier by encoding the entity name and the secret value, wherein ~~such that~~ by using the secret value necessary for access, ~~information necessary to access~~ the cryptographic context information can be retrieved, and wherein the entity identifier when decrypted with the secret value to acquire the entity name provides an index into a data structure for acquiring the cryptographic context information in order to establish a secure connection to the original endpoint, and wherein the cryptographic context information is shared with a plurality of other entities that are members of a group, each member engages ~~capable of engaging~~ in the secure connection by acquiring the cryptographic context information via the entity name that is acquired by decrypting the entity identifier with the secret value, and wherein the secret value is known to each of the members.

2. (Original) The method of claim 1, further comprising the step of passing the entity identifier to at least one subsequently-connected computing entity which seeks to connect to the original endpoint.

3. (Original) The method of claim 2, further comprising the step of decoding the entity identifier using the secret value, thereby determining information necessary to access the cryptographic context information.

4. (Original) The method of claim 3, wherein the step of decoding the entity identifier comprises using the secret value as a key to an encryption algorithm to decrypt the entity identifier.

5. (Original) The method of claim 3, wherein there is at least one other trusted computing entity, the trusted computing entity possessing a trusted entity name, and the decoding step comprises encoding at least one trusted computing entity name and the secret value to produce a computed identifier, and then comparing the computed identifier to the entity identifier to determine if they match.

6. (Original) The method of claim 3, further comprising a deconcatenating step which deconcatenates a random number from the entity identifier prior to the decoding step, and the decoding step uses the random number, a trusted entity name from one of the group of trusted entity names and the secret value to produce a computed identifier and then compares the computed identifier to the entity identifier to determine if they match.

7. (Original) The method of claim 6, wherein the computed identifier and the entity identifier do not match and wherein there is at least one other trusted computing entity, further comprising repeating the decoding step until a match is found or until there are no more trusted computing entities to try.

8. (Original) The method of claim 2, wherein the subsequently-connecting computing entity uses the originally-connected entity name to access the originally-connected entity cryptographic context information, and the subsequently-connecting computing entity uses the originally-connected entity cryptographic context information in a secure connection to the original endpoint.

9. (Currently Amended) The method of claim 1, whereby the creating step comprises using a hash function with an input and an output, said input comprising the entity name and the secret value, said output comprising the entity identifier, and wherein each member has access to the hash function and knows each entity name for each remaining member, which permits each member to serially provide a particular entity name and the secret value to the hash function and compare a result against the entity identifier, the processing continues until a match is found with the entity identifier or until each available entity name has been attempted indicating there is no match.

10. (Currently Amended) The method of claim 1, whereby the creating step comprises using a hash function with an input and an output, said input comprising a bitwise concatenation of the entity name, the secret value, and a random number, said output of the hash function being at least bitwise concatenated with the random number, and wherein each member has access to the hash function and knows each entity name for each remaining member, which permits each member to bitwise de-concatenate the entity identifier to acquire the random number and to use the random number, the secret value, and each entity name with the hash function to serially check for a particular entity name represented in the entity identifier.

11. (Cancelled).

12. (Original) The method of claim 10, wherein the hash function is SHA-1.

13. (Original) The method of claim 1, wherein the creating step comprises using an encrypting algorithm that uses a key to encrypt the entity name using the secret value as the key, the encrypted entity name comprising the entity identifier.

14. (Original) The method of claim 1, wherein the creating step comprises bitwise concatenating the entity name and a random identifier comprising a result and then using an encrypting algorithm that comprises an input, a key, and an output, whereby the result comprises the input, the secret value comprises the key, and the output comprises the entity identifier.

15. (Original) The method of claim 14, wherein the encrypting algorithm is Triple DES.

16. (Original) The method of claim 2, wherein the originally-connected entity is no longer connected to the original endpoint.

17.-18. (Cancelled).

19. (Currently Amended) A system for sharing secure cryptographic connections, the system comprising: an originally-connected trusted entity which comprises an originally-connected entity name and cryptographic context information; at least one other trusted entity, which comprises another entity name; a secret value known to the at least two trusted entities and agreed upon as part of start-up procedures for the at least two trusted entities; and a connection identifier comprising an encoding of the originally-connected entity name and the secret value, which is known to the at least two trusted entities, and wherein the originally-connected entity name encrypted within the connection identifier is linked to the cryptographic context information and provides a mechanism by which a secure connection can be made to the originally-connected trusted entity, and wherein each of the at least two trusted entities can engage in the secure connection by acquiring the cryptographic context information using the secret value, already known to each of the at least two trusted entities, to acquire the originally-connected entity name and then the cryptographic context information and also including ~~The system of claim 18, further comprising~~ a connector which uses the connection identifier to access the originally-connected entity cryptographic context information, and which uses the originally-connected entity cryptographic context information to establish a secure connection to the original endpoint.

20. (Original) The system of claim 19, wherein the originally-connected entity is no longer connected.

21. (Currently Amended) The system of claim [[17]] 19, further comprising a decoder which returns the originally-connected entity name when it is given the connection identifier.

22. (Original) The system of claim 21, wherein the decoder decrypts the connection identifier into an intermediate value when given the secret value and then deconcatenates the originally-connected entity name and the random id from the intermediate value.

23. (Original) The system of claim 21, wherein the decoder deconcatenates the connection identifier into an intermediate value and a random number, and wherein the system further comprises a recoder which recodes the random number, the at least one other trusted entity name, and the secret value into a test identifier.

24. (Original) The system of claim 23, further comprising a tester which compares the connection identifier with the test identifier, and if they are equal determines that the trusted entity name used by the recoder is the originally-connected trusted entity name, and if they are not equal chooses a previously-unchosen trusted entity name as input into the recoder.

25. (Cancelled).

26. (Currently Amended) The system of claim 19, further comprising an encoder which encodes the connection identifier using at least the originally-connected entity name and the secret value, wherein ~~The system of claim 25, whereby~~ the encoder bitwise concatenates the entity name and a random number producing an intermediate value and then uses an encryption algorithm that takes a key to encrypt the intermediate value using the secret value as the key.

27.-31. (Cancelled).

32. (Currently Amended) The system of claim 19, wherein an ~~25, whereby the~~ encoder creates the connection identifier by bitwise concatenating two values; a [[the]] first value being a random number, and a [[the]] second value being the output of a hash function with an input and an output, the input comprising the bitwise concatenation of the entity name, the secret value, and the random number, and wherein the random number is bitwise de-concatenated by each of the at least two trusted entities to acquire the random number, each of the at least two trusted entities then serially provides an entity name for another member along with the random number and the secret value to the hash function to determine which particular entity name is represented in the entity identifier.

33.-34. (Cancelled).

35. (Currently Amended) A configured storage medium embodying data and instructions readable by a computer to perform a method of sharing secure cryptographic connections between trusted computing entities which share a secret value, the computer-implemented method comprising the steps of:

connecting an originally-connected entity to an original endpoint, the originally-connected entity having an entity name and cryptographic context information; and

creating an entity identifier by encoding the entity name and the secret value, ~~such that~~ wherein by using the secret value ~~information~~ necessary ~~to~~ for access, ~~access~~ the cryptographic context information can be retrieved by decrypting the entity identifier to acquire the entity name and using the entity name as an index into a data structure for acquiring the cryptographic context information to establish a secure connection, and wherein members of a group know the secret value and each member can use the secret value to decrypt the entity identifier and acquire the entity name and use the entity name to acquire the cryptographic context information that permits each member of the group to engage in the secure connection with the original endpoint.

36. (Currently Amended) The configured storage medium of claim 35, whereby the creating step comprises using a hash function, and wherein each member knows each entity name for remaining members and has access to the hash function permitting each member to serially check outputs from the hash function for each entity name using that particular entity name and secret value as input until the entity identifier is reproduced.

37. (Original) The configured storage medium of claim 35, wherein the creating step comprises encrypting a bitwise concatenation of the entity name and a random value.